(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(72) Inventor: QUINN, Paul, E.; 41 Ardmore Road, Kensington, CA 94707 (US).

(54) Title: SYSTEM FOR CREATING DYNAMIC GRAPHICAL USER INTERFACES BY THE USE OF COLOR-ENCODED FUNCTIONALITY MAPS

(57) Abstract: User interfaces are designed simply, but with a great deal of flexibility by a set of bitmap graphic files. Color codes are predefined for use in a map graphic that specifies the position, size and functionality of control regions based on a correspondence between the color codes and a set of controls that an application program can generate in the user interface. A body graphic defines the overall appearance of the user interface within the size specified by the map graphic. Control graphics define the appearance of the control regions specified by the map graphic based on a predefined order of supplying different states of the control graphics for the control regions. Internal maps within the files containing control graphics may be used to add additional information regarding the size and shape of graphic elements that are displayed in a control graphic, such as the slider on a volume or gain slider control, or the width of characters in a variable pitch font.

SYSTEM FOR CREATING DYNAMIC GRAPHICAL USER INTER-FACES BY THE USE OF COLOR-ENCODED FUNCTIONALITY MAPS

CROSS-REFERENCE TO RELATED APPLICATION

This application is related to U.S. provisional patent application Serial No. 60/154,321, filed September 17, 1999 entitled METHOD FOR CREATING DYNAMIC GRAPHICAL USER INTERFACES BY THE USE OF COLOR-ENCODED FUNCTIONALITY MAPS, and the nonprovisional U.S. patent

5   application corresponding thereto, entitled SYSTEM FOR CREATING DYNAMIC GRAPHICAL USER INTERFACES BY THE USE OF COLOR-ENCODED FUNCTIONALITY MAPS, filed September 15, 2000, both incorporated herein by reference.

BACKGROUND OF THE INVENTION

10   Field of the Invention

The present invention is directed to generation of graphical user interfaces for a computer and, more particularly, to using color encoded functionality maps to simplify the process of defining a "skin" for computer software.

Description of the Related Art

15   It is now common to be able to customize the user interface of computer programs. Users are often given the capability of selecting what menus or tool bars appear, how they are oriented and how they respond to the operational mode

of the program in, e.g., word processing programs and other application pro-
grams. Some programs are written permitting others to design the appearance of
the user interface. Examples include multimedia players, such as Nullsoft
Winamp and RealJukebox™. However, conventionally the choices of display are
either very limited or require the use of programming techniques that require a
significant degree of effort on the part of the user interface designer.

In the case of user interfaces which are often referred to as "skins," the
appearance of the user interface is often highly artistic. Thus, the design of such
user interfaces requires an individual who has both artistic ability and knowledge
of the computer programming skills necessary to define the user interface, or the
cooperation of two individuals who together have the requisite skills. Winamp
uses a set of what may be called control graphics files with interface elements that
are fixed in size and location on the interface and on the control graphic files
themselves. No variation in the location or size of any of the controls is permit-
ted. RealJukebox™ uses a combination of graphics and text files to produce a
configurable interface. This provides a great deal of design flexibility, but requires
significant more knowledge of how to control a computer. There is no known way
for a graphic artist without computer skills to define a user interface without the
assistance of another person.

SUMMARY OF THE INVENTION

It is an object of this invention to provide a way for graphic artists to input
program development information to a computer using graphic images instead of
text.

It is another object of the present invention to provide a user interface for a
computer program that is defined by graphic images, including how the user
interface responds to user manipulation of an input device.

The above objects can be attained by a method of communicating with a
computer, including conveying program development information through posi-
tions and sizes of colored regions in a graphic image. For example, the program
development information may define a user interface by providing two types of

5

10

15

20

25

30

files: a map file containing a map graphic specifying positions and sizes of at least one control region and at least one information providing region and at least one control file containing at least one graphic image to be displayed in the at least one control region. The user interface is then generated by reading the at least one control file and the map file in which each control region has a predefined color corresponding to a control type, and displaying the interface graphic with each control region replaced by a corresponding control graphic. Preferably, the control file includes first and second control graphics corresponding to the at least one control region, the first control graphic is initially displayed in a corresponding control region, and the second control graphic is displayed in the corresponding control region in response to detection of user manipulation of an input device in relationship to the corresponding control region.

These objects, together with other objects and advantages which will be subsequently apparent, reside in the details of construction and operation as more fully hereinafter described and claimed, reference being had to the accompanying drawings forming a part hereof, wherein like reference numerals refer to like parts throughout.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram of a computer system on which the present invention can be implemented.

FIGS. 2A and 2B are examples of map graphics used by an embodiment of the present invention.

FIGS. 3A and 3B are examples of body graphics used by an embodiment of the present invention.

FIGS. 4A-11C are examples of control files according to the present invention.

FIG. 12 is a flowchart of a method of generating a user interface according to the present invention.

FIGS. 13A and 13B are the resulting user interfaces corresponding to the map graphics illustrated in FIGS. 2A and 2B, respectively.

-4-

DESCRIPTION OF PREFERRED EMBODIMENTS

The present invention could be applied to program development of many kinds of computer applications. The following description of the present invention as specifying the user interface of a multimedia player for a desktop computer

5   should not be construed as limiting the present invention to multimedia players.

A simplified block diagram of a conventional computer system 10 on which the present invention can be implemented is illustrated in Fig. 1. Computer system 10 includes a processor 12 that receives input from one or more input devices 14, such as a keyboard, mouse, trackball, or other pointing device. Proces-

10   sor 10 accesses data and programs in any type of conventional memory unit 16 and generates visible output on a display device 18, such as a computer monitor. When executing a multimedia application, processor 12 may also output audio signals to speakers 20 or external components. For example, computer system 10 may be a personal computer system with an Intel x86 or Pentium®, AMD, or

15   Motorola processor in a desktop, tower or notebook configuration. Any of several conventional operating systems may be stored in memory unit 16, such as Microsoft® Windows®, Apple® OS, etc., capable of executing applications having a graphical interface. The present invention is not limited to the operating systems and hardware described, but can be applied to any known hardware and operating

20   system environment which supports these basic and well known capabilities. The description below will be for computer system 10 operating under Microsoft® Windows®.

According to the present invention, a conventional application program written in e.g., C, C++, or some other procedural or non-procedural computer

25   language is executed to analyze graphic images for information formatted as described below. Coding of the program does not require any unusual skills and therefore only a flowchart of the basic steps performed by the program is provided herein. In the following description, the term "file" will often be used to describe the data structure containing graphic images processed by the application, but any

30   type of input may be use to supply the graphic images, whether by a file stored on

a disk or in memory, received via a communication device, encoded with the program, or provided in any other conventional manner.

The data processed by the program to specify the user interface, commonly called a "skin," include a map graphic or map file that defines the overall geome-
5   try of the graphic image or "body" to be displayed by the application. Predefined colors are used to indicate where control regions and display regions are to be located within the body. Two examples of map graphics are provided in Figs. 2A and 2B to suggest some of the variation that is possible. Both drawings are for a multimedia player capable of playing compact discs (CD) and digital audio files.
10   In both Figs. 2A and 2B, the color black indicates main body 30, the gray colors indicate extended area(s) 32, the green colors indicate timer control regions and skin mode regions 34, the turquoise colors indicate text display regions 36, the red colors indicate CD control regions 38, the yellow colors indicate slider control regions 40, the blue colors indicate window control regions 42 and the violet
15   colors indicate track box control regions 44 and display regions 46.

A list of the colors available for the illustrated embodiment is provided in the Table below containing hexadecimal and decimal values in a 256 color system and what each color represents. The application program has access to a map table corresponding to the following table. The map table may be included as part
20   of the program, or inputted in the same manner as the graphics files. The Table below could be used to generate color graphics files with the size and shape of the control regions in the drawings.

TABLE

Hexadecimal	Decimal	Control Region
#999900	(153,153,0)	Volume 1
#CCCC00	(204,204,0)	Volume 2
#FFFF33	(255,255,51)	Bass Adjust
#FFFF99	(255,255,153)	Treble Adjust
#FFFFCC	(255,255,204)	Balance Adjust
#FFFF00	(255,255,0)	Track Progress Bar

TABLE (cont.)

#CCCC66	(204,204,102)	Visual Effects Level/Gain
#666600	(102,102,0)	Playlist Scroll
#0033CC	(0,51,204)	Local Library/Database
#003366	(0,51,102)	Edit Track Info
#003399	(0,51,153)	Extended Info
#336699	(51,102,153)	Open Detachable Ext Area
#6666FF	(102,102,255)	Open Ext Area 1
#6666CC	(102,102,204)	Open Ext Area 2
#666699	(102,102,153)	Open Ext Area 3
#99CCIFF	(153,204,255)	CDDB Mini#Browser
#3333FF	(51,51,255)	CDDB Music Links
#000099	(0,0,153)	Menu
#0000FF	(0,0,255)	Close
#0000CC	(0,0,204)	Minimize
#CCCFF	(204,204,255)	Visual Effects Area
#9999FF	(153,153,255)	Next Visual Effect
#9966FF	(153,102,255)	Detach Vis Area
#33CCFF	(51,204,255)	QCD Web Site Link
#996699	(153,102,153)	Add Tracks to Playlist
#66DD66	(102,0,102)	Delete Tracks
#CC66CC	(204,102,204)	Sort Tracks
#FF66FF	(255,102,255)	Save Playlist
#FF99FF	(255,153,255)	Repeat Current Track
#FF33FF	(255,51,255)	Repeat Playlist
#FFCCFF	(255,204,255)	Shuffle Playlist Order
#CC00CC	(204,0,204)	Scroll Playlist Box Up
#990099	(153,0,153)	Scroll Playlist Box Down
#FF00FF	(255,0,255)	Track Area
#000000	(0,0,0)	Main Window Body
#CCCCC	(204,204,204)	Extended Area 1
#999999	(153,153,153)	Extended Area 2

TABLE (cont.)

	#999999	(153,153,153)	Extended Area 2
	#666666	(102,102,102)	Extended Area 3
	#333333	(51,51,51)	Extended Area 4 (detachable)
5	#33FF33	(51,255,51)	Track Time
	#99FF99	(153,255,153)	Playlist Time
	#009900	(0,153,0)	Elapsed Time
	#00CC00	(0,204,0)	Remaining Time
	#00FF00	(0,255,0)	Time Digit Area
10	#00CCCC	(0,204,204)	Artist
	#00FFFF	(0,255,255)	Album
	#009999	(0,153,153)	Track
	#66FFFF	(102,255,255)	System Message
	#99FFFF	(153,255,255)	Browser Message
15	#660000	(102,0,0)	Eject
	#990000	(153,0,0)	Previous Track
	#CC0000	(204,0,0)	Next Track
	#FF0000	(255,0,0)	Stop
	#FF3333	(255,51,51)	Pause
20	#FF6666	(255,102,102)	Play
	#33FF00	(51,255,0)	Next Skin Mode
	#00FF11	(0,255,17)	Skin Mode 1
	#00FF22	(0,255,34)	Skin Mode 2
	#00FF33	(0,255,51)	Skin Mode 3
25	#00FF44	(0,255,68)	Skin Mode 4
	#00FF55	(0,255,85)	Skin Mode 5
	#00FF66	(0,255,102)	Skin Mode 6
	#00FF77	(0,255,119)	Skin Mode 7
	#00FF88	(0,255,136)	Skin Mode 8
30	#00FF99	(0,255,153)	Skin Mode 9

To generate the user interface, the application program reads or scans the map graphic to identify the pixels that match the colors in the map table. Thus, the map graphic specifies the positions and sizes of the control regions and display regions in the body that will be generated for the user interface. Since the application program determines the location and size of the control regions based on color, it is necessary for the designer of the user interface to use a program that creates precise colors. Some graphics programs have an anti-aliasing feature that modifies the colors at the borders of an area to improve the appearance of the graphic on a display screen. Anti-aliasing will modify the color values at applied areas, thus modifying the meaning of the map's color values at those locations. It is essential that the anti-aliasing feature be turned off when creating bit map files for use by an application that implements the present invention.

As illustrated in Figs. 2A and 2B, the map graphic also defines the shape of the body of the user interface, including the main body 30 which is displayed at all times and an extended area 32, that is not permanently displayed. In the illustrated embodiment, the main body 30 would be black, as indicated in the Table. However, in Figs. 2A and 2B, the main body 30 has not been shown as black, so that the connections between the reference numerals and the control regions can be seen. As described in more detail below, the extended area 32 of the body is displayed only when a user manipulates input device 14 in relationship to a control region in the main body, e.g., by depressing a mouse button while a mouse cursor is located over an extend button 48 in the main body.

As indicated in the Table, in the embodiment illustrated in Figs. 2A and 2B the extended area 32 is defined as the area having the color in Windows 256 level RGB of (204, 204, 204). As also indicated in Table, the example illustrated in Fig. 2B may include three additional extended areas in which all three of the colors, red, green and blue are either 153, 102 or 51. According to both examples, the control regions are permitted to extend to the edge of the extended area 32 by defining the extended area 32 as including any control that touches an extended area pixel, other than the extend button(s) 48.

-9-

The map graphics illustrated in Figs. 2A and 2B each include a rectangular display region 46. Like any of the other regions, the display region 46 does not have to be rectangular, but can have any shape. Unlike the control regions defined by the map graphic, a display region does not include any interactive capability by the application program generating the user interface. However, the display region may be used by another program, e.g., a "plug-in" which includes controls with which a user can interact.

The appearance of the body of the user interface is defined by a body graphic. Examples of body graphics corresponding to the map graphics illustrated in FIGS. 2A and 2B are illustrated in FIGS. 3A and 3B. Although several of the control buttons are shown in FIGS. 3A and 3B, these are provided for the benefit of the designer and will be replaced by the application program as described below. Only the portion of the body graphic corresponding to the black portions of FIGS. 2A and 2B and the light gray disc portions to the right thereof are displayed by the application program.

A set of graphics, e.g., in five bitmap files, define the characteristics of what is displayed in control regions. In the preferred embodiment, any of these files or portions thereof described below may be excluded, provided the corresponding colors are not present in the map graphic. If there is no extend button 48 defined in the map graphic, and an extended area 32 is included in the map graphic, the extended area 32 will remain displayed at all times. Thus, the only difference between the controls in the extended area 32 and in the main body 30 would be that an extend button 48 can be added later by the designer to hide the controls in the extended area 32.

In the examples illustrated in Figs. 2A and 2B, the control graphics for the basic player controls, like play, stop, pause, etc. are supplied in a file named ButtonSet.bmp. The basic player control graphics for the map graphics illustrated in Figs. 2A and 2B are illustrated in Figs. 4A and 4B, respectively. As apparent from Figs. 4A and 4B, as well as the corresponding red regions 38 in the map graphics of Figs. 2A and 2B, among the shapes that buttons may take are rectangles, ovals, etc.

-10-

The control graphic files each contain a single graphic which is read by the application program to obtain the "control graphics" that correspond to the control regions defined by the map graphic. Each control graphic is repeated four times in each row illustrated in Figs. 4A and 4B, because in this embodiment the

5   application program expects to find four control graphics in each row of the ButtonSet.bmp file arranged in a pre-defined manner. Each row of control graphics corresponds to a different control region and is separated from the next row of control graphics by one pixel. If control region(s) are not defined in the map graphic, the application program generating the user interface skips to the

10   next defined control region in the predefined order.

For example, in the embodiment illustrated in FIGS. 4A and 4B, the first row of ButtonSet.bmp defines the appearance of the eject button and the following rows define the appearance of the track backward, track forward, stop, pause and play buttons. The first button in each row defines the normal appearance of that

15   button when the mouse cursor is not within the corresponding control region and the following graphic elements define the appearance of the control region when the mouse cursor is within the control region but no button is depressed, when the button is depressed and the mouse cursor is off (in a manner similar to what are commonly termed "radio buttons"), e.g., to indicate a pressed, but not active state,

20   and when the mouse cursor is within the control region and a mouse button is depressed. Thus, the ButtonSet.bmp file defines the characteristics of five control regions corresponding to four different states of user manipulation of an input device, such as a mouse, for each of the control regions. By making one state or graphic element the same as another, e.g., the second and third states, it will

25   appear to the user that the button cannot have a state, such as "depressed and inactive" which may be desirable to a designer of a user interface.

An application program according to the illustrated embodiment uses similar rules to define the characteristics of other control regions by reading graphics like those illustrated in Figs. 5A-8B. Figs. 5A and 5B provide examples

30   of control graphics associated with windows buttons and show that not all types of control regions that could be defined for an application program are required for a

particular "skin". In the examples illustrated in Figs. 5A and 5B, the first row of graphic elements corresponds to the menu button which opens a menu for the application program, followed by minimize and close buttons for the window displaying the body of the user interface for the application program. Both Figs.

5   5A and 5B include an extend button 48a for the first extended area. However, in Fig. 5B the extend button 48a is followed by extend button 48b for the second extended area and extend button 48c for the fourth extended area that are not defined in Fig. 5A. Next in both Figs. 5A and 5B are buttons for a menu of links and visualization change. Following in Fig. 5B only are buttons for detaching the

10   visualization area, editing the current song, obtaining extended information, and accessing the local music library. The next to last row of Figs. 5A and 5B defines the appearance of a button activating a CDDB music browser which may be a separate application. The last row of Figs. 5A and 5B opens a web browser to access a website associated with the application program.

15   In the example illustrated in Fig. 5B, the first, second and fourth extended areas display mixer controls, the button illustrated in the last row of Fig. 5B, and a playlist, respectively. In this example, the links menu button displays a menu of music-related URLs supplied by a website, the visualization buttons change the animated graphic displayed in region 46 and permit display region 46 to be

20   detached from the body of the user interface. The edit song button brings up a window for displaying song and track information and permitting editing thereof. The extended information button brings up a window that links to an Internet database displaying information related to the music being played. The music library button displays a menu for accessing a local database of information related

25   to music files that have been accessed by the player application.

        Some of the control graphic files include an internal map graphic, or simply an internal map, that has a function similar to that of the map graphic illustrated in Figs. 2A and 2B, for the graphic elements included in that particular control graphic file. Internal maps are scanned the same way as the map graphic

30   file to obtain further information about a particular graphic element within the control region. The first row of the TrackSet.bmp files illustrated in Figs. 6A and

-12-

6B provides examples of internal maps that according to the Table would be a magenta color. The internal maps indicate to the application program the size of a track button inside the trackbox area in the map graphic, and in Fig. 6A, an internal map (that would be blue according to the Table) for track numerals that are displayed instead of the text name of the track, as displayed in the example illustrated in Fig. 6B. This extra color information (from the internal map) is required due to the restrictions imposed by the two-dimensional nature of the map graphic. Only one color (i.e., one piece of information) per pixel location can be indicated by the map graphic. In general, internal maps are required to give further information regarding a control region where an overlap in colors would occur if the information was attempted to be provided by the map graphic.

The graphic elements in each row typically indicate "normal", "mouse over", "mouse over and button down", and "mouse off and button down". Some of the rows include a graphic element indicating "marked" for further operations, such as deleting. The application program expects the rows of Figs. 6A to define the following mouse states: normal, mouse-over, depressed mouse-up, depressed mouse-down, and marked (for a suppressed track). In the rows following the internal map in Fig. 6B, the graphic elements indicate the following mouse states: normal, mouse-over, being dragged (to rearrange the order of tracks in a playlist), track selected, and track marked to suppress playback.

The next two rows in the example illustrated in Fig. 6A show the "normal" numerals and the "mouse-over" track numerals, respectively. In the example illustrated in Fig. 6B, track numerals are not used separately, but of course may be included in the text displayed within the text box defined by the internal map on row 1. The next two rows in both Figs. 6A and 6B show examples of scroll up and scroll down buttons when all of the tracks are not displayed with the graphic elements in each row corresponding to the same mouse states as described above with respect to Figs. 5A and 5B. The next four rows in Figs. 6A and 6B respectively correspond to delete track, add track, save playlist and sort playlist. The last three rows in Figs. 6A and 6B correspond to repeating all tracks on the CD or

-13-

in the playlist, repeating the currently playing track and shuffling the tracks, with the same four mouse states defined across each row, as described previously.

Examples of TimerSet.bmp files are illustrated in Figs. 7A and 7B. Like TrackSet.bmp, the first row contains an internal map for individual numerals. The map graphic illustrated in both examples of Figs. 2A and 2B is sufficient large to hold characters indicating preferably at least two minutes digits and two seconds digits, as well as a colon separating the two. In the case of the embodiments illustrated in Figs. 7A and 7B, there is also room for a negative sign, so that the application may be configured to show a decreasing timer. The following graphic elements in Figs. 7A and 7B respectively indicate whether the information displayed represents the current track, or playlist, and an indicator of whether the time that is displayed indicates elapsed or remaining time.

In the examples illustrated in the drawings, several of the control regions have slider controls for controlling and indicating, e.g., track position, volume, balance, frequency level (e.g., base and treble), visualization gain, text scrolling (e.g., for playlists), etc. In the embodiment for the examples illustrated in Figs. 8A and 8B, the slider controls may be defined as tracking, e.g., horizontal, vertical, or diagonal; rotational; or progressive (i.e., "thermometer"). Internal maps are provided in the first row of Figs. 8A and 8B using the same color codes as in the map graphic. If an internal map of the appropriate color, e.g., the rectangle in the upper left hand corner of Figs. 8A and 8B (that would be yellow according to the Table), that fits within the control region defined in the map graphic is provided in the top row, the application program will look for a tracking slider control formed of a "handle" or slider element having the shape defined by the internal map and a slot image which may also be provided in the body graphic. In Fig. 8A, rows 4-6 are examples of diagonal sliders for CD volume, bass and treble and row 7 is a horizontal slider for balance.

If a one pixel high internal map that is wider than the control region defined in the map graphic is provided in the top row of the slider control graphic, the application program will look at the predefined row for a series of graphic elements or frames that fit within the control region defined in the map graphic

and that extend for the length of the internal map of the corresponding color. Examples of rotational controls are illustrated in the third row of Figs. 8A and 8B and in rows 7 and 8 of Fig. 8B. In the examples illustrated in Figs. 8A and 8B, the third row is a master volume slider of a rotational type in which a group of

5   dots arranged in a 180° arc become increasingly brighter sequentially as the volume is increased. A similar rotational slider for visualization gain is provided in row 8 of Fig. 8B. Row 7 of Fig. 8B is a variation of a rotational control for balance.

Since the rotational nature of the control graphics in Figs. 8A and 8B is not

10   easily comprehended, another example is provided by Fig. 8C which has three rotational controls for, e.g., volume, balance and gain of the music visualization region. Each of the last three rows in Fig. 8C begins with the internal map for the control and follows with depictions of the control regions for different levels, because the embodiment of the application program for the example illustrated in

15   Fig. 8C expected to find an internal map at the beginning of each row, instead of all internal maps in the first row. The granularity of the display is determined by the number of graphic elements in a row. In the embodiments illustrated in Figs. 8A-8C, although the display will generate what appears to be a rotating control, changes in level are input by a user sliding the mouse horizontally or vertically

20   after depressing the mouse button while the mouse cursor is over the control region associated with the control graphic.

A progressive or "thermometer-type" control is defined by not providing an internal map for a slider control defined in the map graphic. The application program then looks for a single graphic element in the appropriate row that will be

25   displayed, to a degree corresponding to the level of the control, in place of the image in the body graphic at the location of the corresponding control region in the map graphic. In Fig. 8B, rows 4-6 are progressive sliders for CD volume, bass and treble. A portion of the graphic element from the bottom up (in this example since they are vertically biased) is displayed to represent the level of the control.

30   The portion of the control region above the control's level is obtained from the body graphic.

Figs. 9A and 9B are examples of control graphics used for a banner image that provides a title bar on menus displayed by the application program. The size of the menus may be defined by the program, an internal map, another file similar to the map graphic, etc. In the example illustrated in Fig. 9B, the width of the

5   banner is fixed at 23 pixels, but the height of the banners is defined by the graphic elements themselves, although both must be the same height. The examples illustrated in Fig. 9A has gradations, but the size of the menus may vary. Therefore, in the embodiment of the invention that utilizes the examples illustrated in Fig. 9A, the last twenty rows of pixels are repeated if the menu extends beyond the

10   length of the banner bitmap.

In addition to using fixed shapes to define control and display regions, the present invention may use graphics files to define the appearance of resizable areas. Associated with the example illustrated in Fig. 2B are three resizable windows that can be displayed by the application program: a browser for acces-

15   sing an Internet music database, like that provided by CDDB®, a detached visualization display region, and a preview skin window if the application program is able to access additional skins in a directory or via a network connection. Borders for these resizable windows are defined in a file of graphic elements named BorderSet.bmp which are illustrated in Fig. 10. In this example, each border is

20   separated into 16 segments, four corners and three segments for each of the four sides. The corners and two of the segments on each of the sides do not change when a window is resized, while the "middle" graphic element stretches vertically or horizontally. What is unusual about BorderSet.bmp is that it uses only its internal map to determine its layout, and is independent of the map graphic.

25   However, it still uses assigned colors to define the displayed graphic.

In the examples illustrated in the drawings, text, such as track names and album titles, may be displayed in fonts defined in one of two ways. The designer of the user interface may supply his or her own character set(s) to be used by supplying a bitmap file like that illustrated in Figs. 11A-11C. The file

30   CharSet.bmp illustrated in Figs. 11A-11C uses a type of internal map graphic. The first row of the bitmap has white and non-white pixels for the extent of the

bitmap. Each non-white pixel defines the extent of the character graphic below it and the start of the next character in the bitmap. The internal map allows for variable-width character sets to be used in the skin, based solely on graphic information. Several user-defined fonts may be provided by supplying multiple files of

5   character sets to be used in different display regions or control regions, e.g., by using predefined file names, such as ChrSetDisc.bmp, ChrSetTrack.bmp, etc. Alternatively, the character set(s) may be defined using the conventional way in which applications specify fonts to be used to the operating system, instead of using graphics files according to the present invention.

10   When the graphics and any text files, like ChrSet.ini, used to define the user interface are supplied to an application program created according to the present invention, the application program executes the steps illustrated in Fig. 12 to generate a user interface. The manual steps of determining 100 a color-to-control map table, such as that indicated in the Table, and creating 102 a color-

15   coded map graphic are performed first. The application program reads or scans 104 the map graphic and locates the control regions based on the colors in the map table. The color areas detected 104 in the map graphic are used to specify 106 the position, size and functionality of control regions in the user interface. Control graphics are obtained 108 corresponding to all of the control regions. As indicated

20   in Fig. 12, the control graphics may be obtained by reading files stored along with or inside the application program, or may be obtained dynamically by accessing files stored elsewhere on a computer coupled to computer system 10 (Fig. 1) e.g., via the Internet, or generated by the program, such as by a randomized or otherwise modified color of one or more control graphics. After the map graphic and

25   control graphics have been obtained, a window containing the user interface is generated 110. The application program may permit the user, or some other trigger, to change 112 the user interface, e.g., by selecting a different "skin", and if no changes are made, the user interacts 114 in the manner described above. The end result of the user interfaces having the map graphics illustrated in Figs. 2A

30   and 2B are illustrated in Figs. 13A and 13B.

Preferably the graphic files defining a "skin" are archived together using a program that does not permit the user to make modifications to the skins, e.g., by using a non-standard compression technique, encryption, or any other known method. To make it possible for the user to switch between skins easily, preferably

5   a family of skins are archived together. By providing the application program with the ability to add and delete from an archive, as well as the ability to retrieve skins from the archive, additional skins or skin families may be added or existing ones deleted.

The many features and advantages of the present invention are apparent

10   from the detailed specification and thus, it is intended by the appended claims to cover all such features and advantages of the system which fall within the true spirit and scope of the invention. Further, numerous modifications and changes will readily occur to those skilled in the art from the disclosure of this invention. For example, any user-created set of discrete functions could be represented using

15   the invention, such as formatting and layout of a data display for any type of program. It is not desired to limit the invention to the exact construction and operation illustrated and described; accordingly, suitable modification and equivalents may be resorted to, as falling within the scope and spirit of the invention.

CLAIMS

What is claimed is:

1. A method of communicating with a computer, comprising:

5   conveying program development information through positions and
sizes of colored regions in a graphic image.

2. A method as recited in claim 1,

wherein said conveying provides the program development
information to define a user interface.

10   3. A method as recited in claim 2,

wherein said conveying includes specifying a position, size and
functionality of at least one region in a map graphic.

4. A method as recited in claim 3,

wherein said conveying includes specifying the position and size of
15   at least one control region in the map graphic.

5. A method as recited in claim 4,

wherein said conveying further includes specifying the position and
size of at least one display region in the map graphic.

6. A method as recited in claim 4,

wherein said conveying further includes defining characteristics of
the at least one control region.

7. A method as recited in claim 4,

wherein said defining includes storing at least one file containing at
least one control graphic to be displayed in the at least one control region.

8. A method as recited in claim 7,

wherein the at least one file includes a plurality of control graphics
to be displayed in the at least one control region, the control graphics having a
predefined correspondence to different states of user manipulation of an input
device.

9. A method as recited in claim 7, wherein said specifying further
specifies an extended area of the map graphic for display only in response to user
manipulation of an input device.

10. A method as recited in claim 9,

wherein said specifying includes indicating that at least one
additional region for at least one of control and information display is located in
the extended area.

11. A method as recited in claim 7, wherein the at least one file includes
an internal map graphic specifying a size of at least one graphic element also
included in the at least one file.

12. A method as recited in claim 11, wherein said specifying includes
defining a control region type by whether the internal map graphic for a
corresponding control region is present.

13. A method as recited in claim 11, wherein said specifying includes defining a control region type by whether the internal map graphic is larger than the corresponding control region.

14. A method of generating a user interface of a computer system,
comprising:

     reading a map graphic having at least one control region, each control region having a predefined color corresponding to a control type and at least one control file containing at least one control graphic corresponding to the at least one control region; and

     displaying a body graphic having a shape matching the map graphic, with each portion of the body graphic having a corresponding control region in the map graphic replaced by a corresponding control graphic.

15. A method as recited in claim 14,

     wherein said reading obtains at least first and second control graphics related to the corresponding control region,

     wherein said displaying displays the first control graphic in the corresponding control region, and

     wherein said method further comprises modifying the user interface to display the second control-graphic in the corresponding control region in response to detection of user manipulation of an input device in relationship to the corresponding control region.

16. A method as recited in claim 14,

     wherein said reading detects a main body and at least one extended area, outside the main body, included in the body graphic, each having a size defined by the map graphic, and at least one extended control region defined by the map graphic within the main body, each extended control region having a corresponding extended area, and

     wherein said method further comprises:

initially displaying only the main body; and

subsequently displaying the corresponding extended area upon detection of a user input related to the at least one extended control region.

17. A method as recited in claim 14, wherein the at least one control file includes an internal map graphic specifying a size of at least one graphic element also included in the at least one control file.

18. A method as recited in claim 17, wherein said specifying includes defining a control region type by whether the internal map graphic for a corresponding control region is present.

19. A method as recited in claim 17, wherein said specifying includes defining a control region type by whether the internal map graphic is larger than the corresponding control region.

20. A computer readable medium storing at least one file comprising:

at least one body graphic having at least one control region, each control region having a predefined color corresponding to a control type; and

at least one control file containing at least one control graphic corresponding to the at least one control region.

21. A computer readable medium as recited in claim 20, wherein the at least one file includes a plurality of control graphics corresponding to a single control region in the body graphic, each of the control graphics representing different states of user manipulation of an input device.

22. A computer readable medium storing at least one program to control a computer to perform a method of generating a user interface, said method comprising:

reading a map graphic having at least one control region, each control region having a predefined color corresponding to a control type and at least one control file containing at least one control graphic corresponding to the at least one control region; and

5   displaying a body graphic having a shape matching the map graphic, with each portion of the body graphic having a corresponding control region in the map graphic replaced by a corresponding control graphic.

23. A computer readable medium as recited in claim 22,

wherein said reading obtains at least first and second control graphics related to the corresponding control region,

wherein said displaying displays the first control graphic in the

5   corresponding control region, and

wherein said method further comprises modifying the user interface to display the second control graphic in the corresponding control region in response to detection of user manipulation of an input device in relationship to the corresponding control region.

24. A computer readable medium as recited in claim 22,

wherein said reading detects a main body and at least one extended area, outside the main body, included in the body graphic, each having a size defined by the map graphic, and at least one extended control region defined by

5   the map graphic within the main body, each extended control region having a corresponding extended area, and

wherein said method further comprises:

initially displaying only the main body; and

subsequently displaying the corresponding extended area

10   upon detection of a user input related to the at least one extended control region.

25. A computer readable medium as recited in claim 22, wherein the at least one control file includes an internal map graphic specifying a size of at least one graphic element also included in the at least one control file.

26. A computer readable medium as recited in claim 25, wherein said
5   specifying includes defining a control region type by whether the internal map graphic for a corresponding control region is present.

27. A computer readable medium as recited in claim 25, wherein said specifying includes defining a control region type by whether the internal map graphic is larger than the corresponding control region.

10   28. A computer, comprising:
            a memory to store a map graphic having at least one control region, each control region having a predefined color corresponding to a control type and at least one control file containing at least one control graphic corresponding to the at least one control region;
15   a display unit; and
            a processor, coupled to said memory and said display unit, to generate on said display unit a display of a body graphic having a shape matching the map graphic, with each portion of the body graphic having a corresponding control region in the map graphic replaced by a corresponding control graphic.

29. A computer as recited in claim 28,
            further comprising an input device coupled to said processor,
            wherein said memory further stores in the at least one control file at least first and second control graphics related to the corresponding control region,
5   and
            wherein said processor controls the display unit to initially display the first control graphic in the corresponding control region, and to display the second control graphic in the corresponding control region in response to detection

of user manipulation of said input device in relationship to the corresponding

control region.

30. A computer, comprising:

memory means for storing a map graphic having at least one control

region, each control region having a predefined color corresponding to a control

type and at least one control file containing at least one control graphic

5   corresponding to the at least one control region;

display means for displaying a user interface; and

processor means for generating on said display means a display of a

body graphic having a shape matching the map graphic, with each portion of the

body graphic having a corresponding control region in the map graphic replaced

10   by a corresponding control graphic.
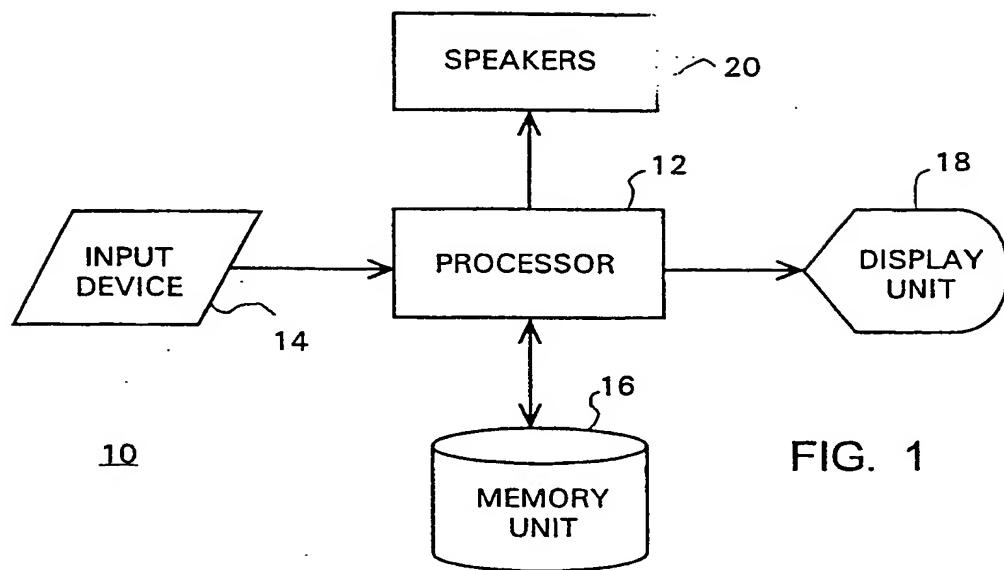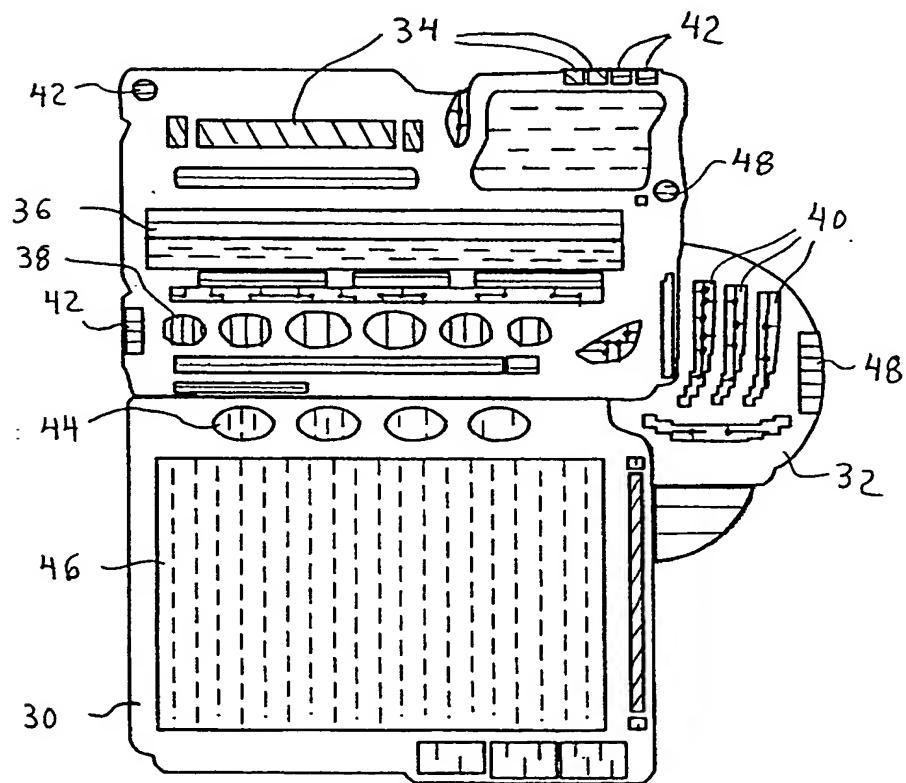
FIG. 1

FIG. 2B
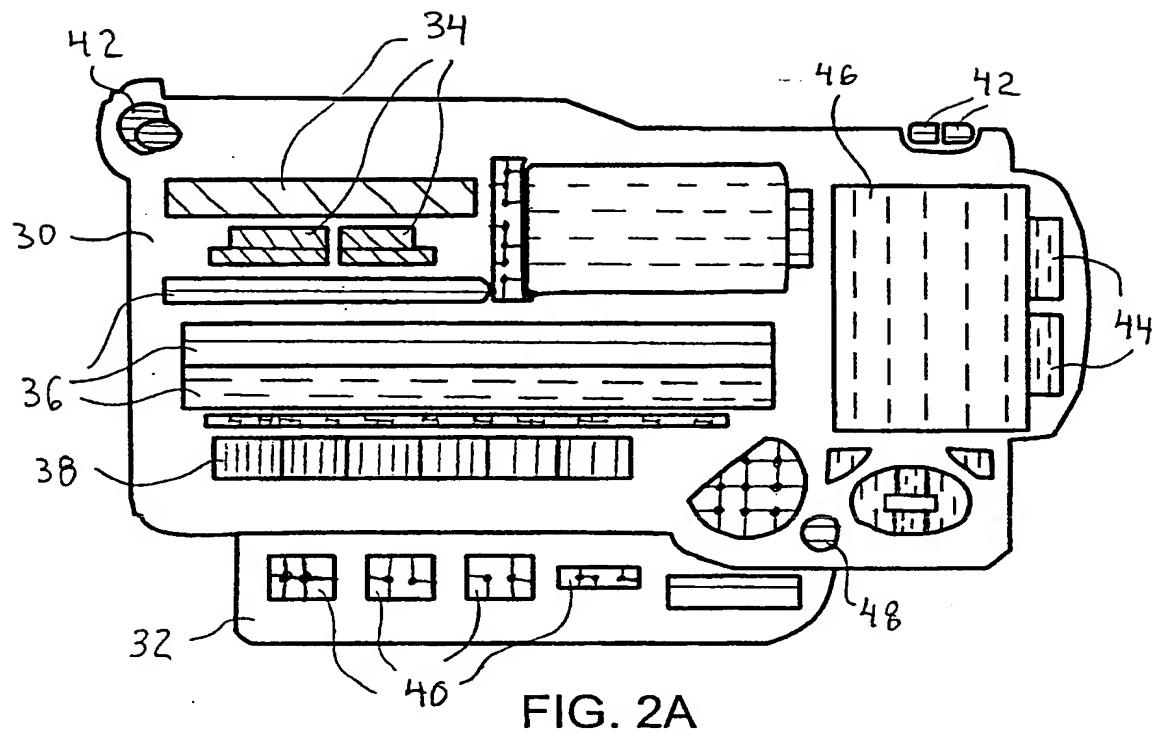
FIG. 2A
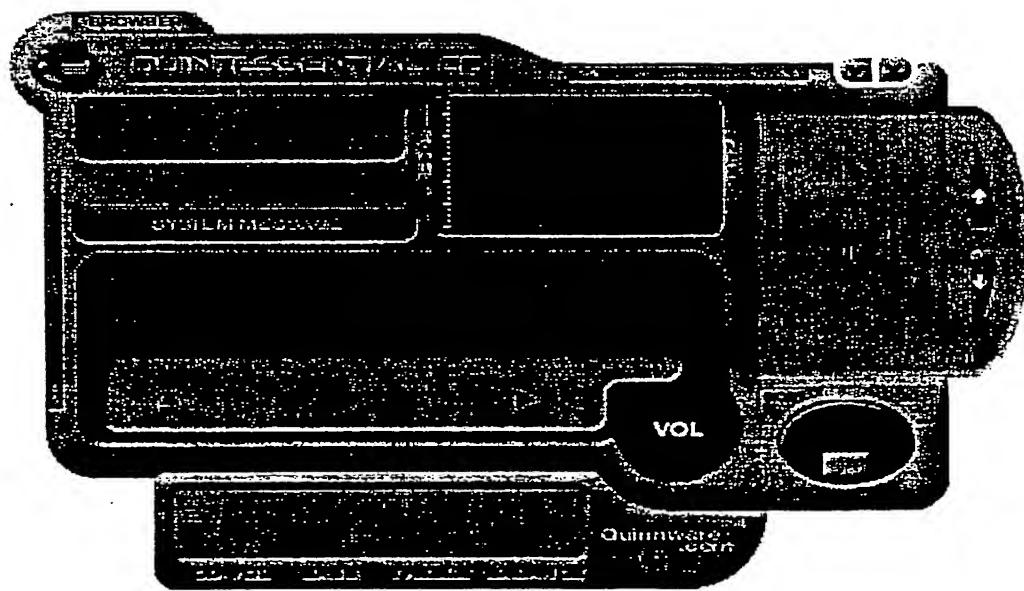
FIG. 3A
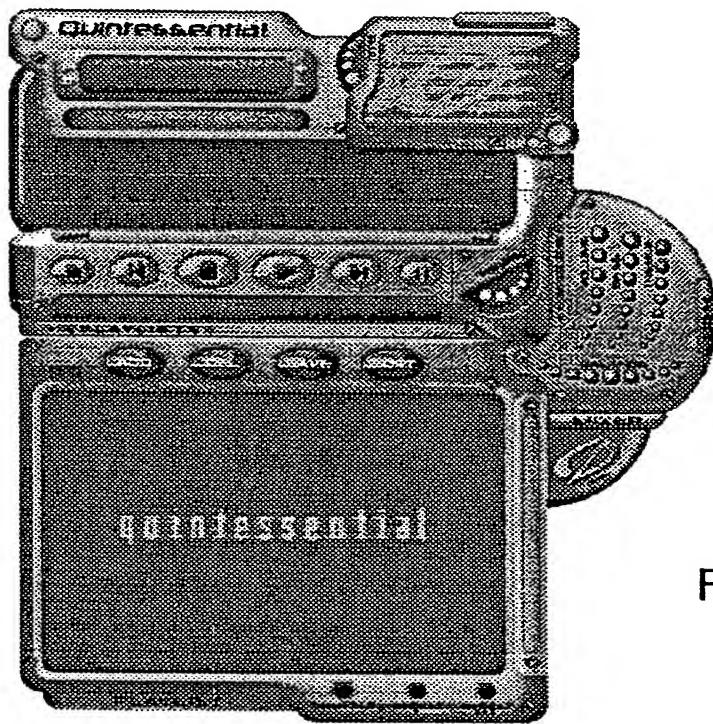
FIG. 3B
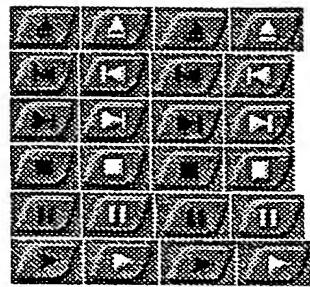
FIG. 4A
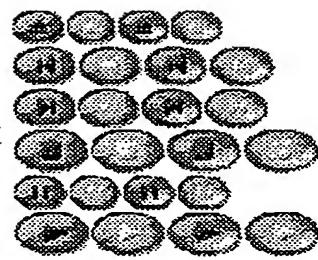
FIG. 4B
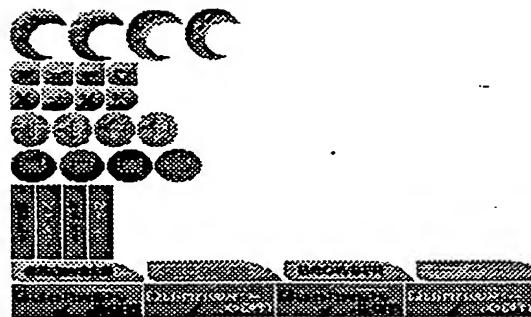
FIG. 5A
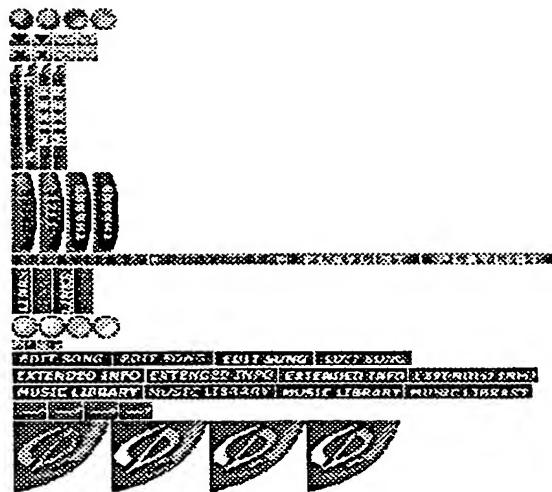
FIG. 5B
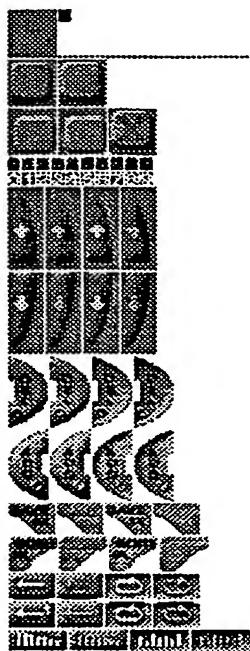
FIG. 6A
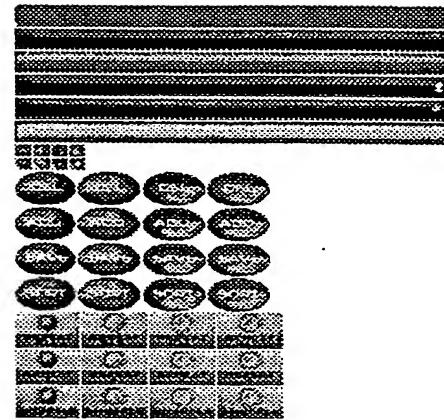
FIG. 6B
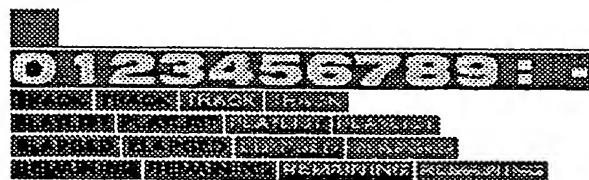
FIG. 7A

FIG. 7B
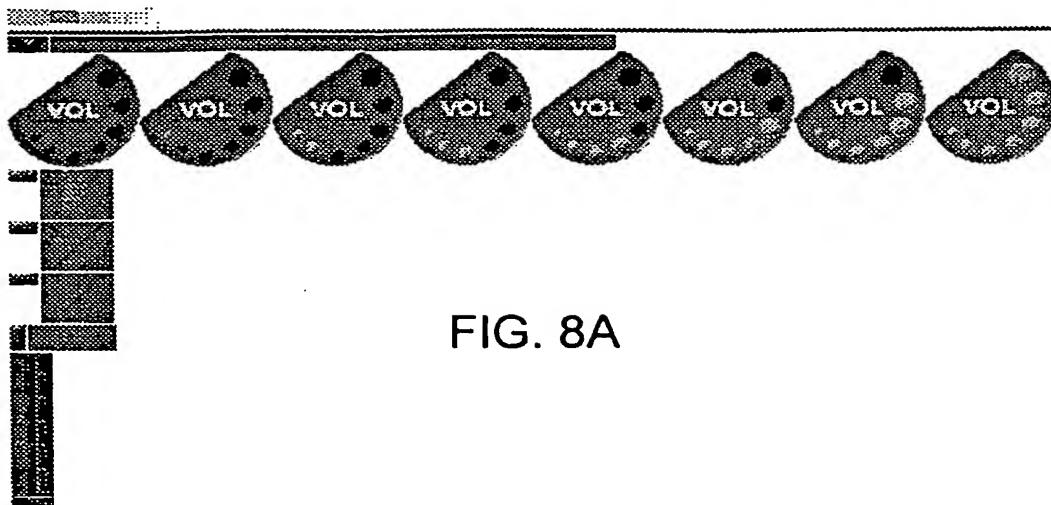
FIG. 8A
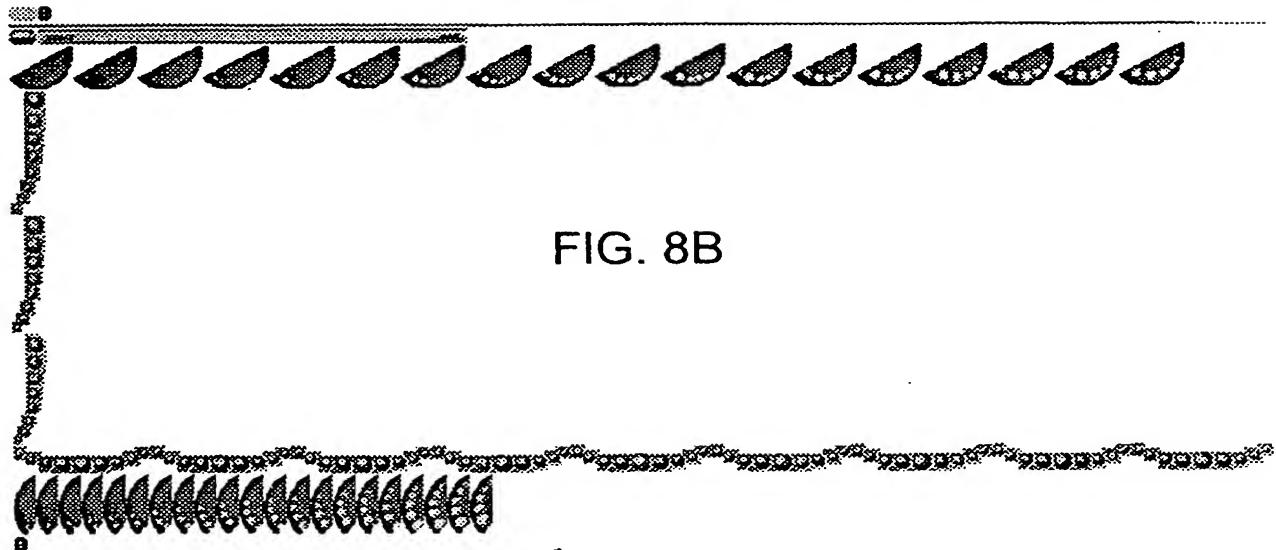
FIG. 8B
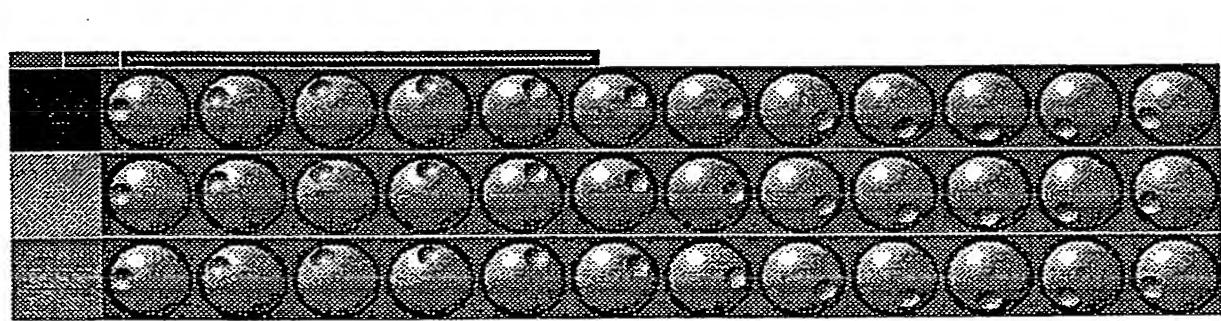
FIG. 8C
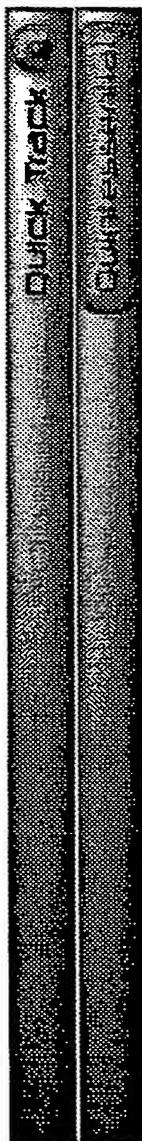
FIG. 9A
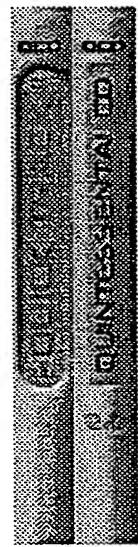
FIG. 9B
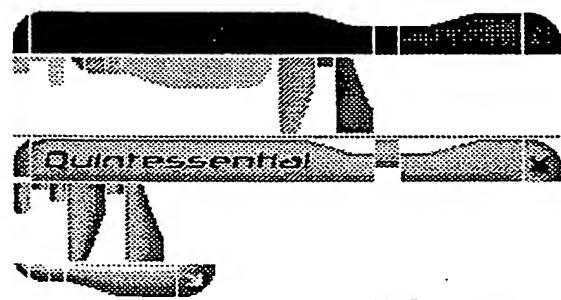
FIG. 10

!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`

FIG. 11A

abcdefghijklmnopqrstuvwxyz{|}~—'''""•–—ˆ˜‚ƒ„…†‡‰‹›™¡¢£¤¥¦§¨©ª«¬®¯°±²³´µ¶·

FIG. 11B

¸¹º»¼½¾¿ÀÁÂÃÄÅÆÇÈÉÊËÌÍÎÏÐÑÒÓÔÕÖ×ØÙÚÛÜÝÞßàáâãäåæçèéêëìíîïðñòóôõö÷øùúûüýþÿ

FIG. 11C

100

DETERMINE
COLOR-TO-
CONTROL
MAP TABLE

CREATE COLOR-CODED
MAP GRAPHIC   102

FIG. 12

104

SCAN MAP GRAPHIC FOR
ALL COLORED AREAS

USE COLOR AREAS TO
DETERMINE POSITION AND
SIZE OF CONTROL REGIONS   106

OBTAIN CONTROL GRAPHICS
STATICALLY OR
DYNAMICALLY GENERATE
GRAPHICS TO FILL CONTROL
REGIONS   108

110

DRAW WINDOW AND CONTROLS AS
LAID OUT BY COLOR-CODED MAP
GRAPHIC AND FILL WITH APPROPRIATE
CONTROL GRAPHICS

114

USER INTERACTS
WITH WINDOW AND
CONTROLS.
CONTROL REGIONS
UPDATED
ACCORDINGLY.

112

YES   CHANGE
             USER
             INTERFACE?   NO

FIG. 13A

FIG. 13B

(54) Title: SYSTEM FOR CREATING DYNAMIC GRAPHICAL USER INTERFACES

(57) Abstract: User interfaces are designed simply, but with a great deal of flexibility by a set of bitmap graphic files. Color codes are predefined for use in a map graphic that specifies the position, size and functionality of control regions based on a correspondence between the color codes and a set of controls that an application program can generate in the user interface. A body graphic defines the overall appearance of the user interface within the size specified by the map graphic. Control graphics define the appearance of the control regions specified by the map graphic based on a predefined order of supplying different states of the control graphics for the control regions. Internal maps within the files containing control graphics may be used to add additional information regarding the size and shape of graphic elements that are displayed in a control graphic, such as the slider on a volume or gain slider control, or the width of characters in a variable pitch font.

INTERNATIONAL SEARCH REPORT

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7   G06F9/44

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7   G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal, WPI Data, IBM-TDB

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	NCSA HTTPD DEVELOPMENT TEAM: "NCSA IMAGEMAP TUTORIAL" , 'Online! 11 May 1995 (1995-05-11), pages 1-5, XP002163997 Retrieved from the Internet: <URL:http://hoohoo.ncsa.uiuc.edu/docs/tutorials/imagemapping.html> 'retrieved on 2001-09-10!	1-7, 11-14, 17-20, 22, 25-28,30
Y	page 1, last line -page 4, line 12 --- -/--	8-10,15, 16,21, 23,24,29

[X] Further documents are listed in the continuation of box C.   [X] Patent family members are listed in annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not
     considered to be of particular relevance
"E" earlier document but published on or after the international
     filing date
"L" document which may throw doubts on priority claim(s) or
     which is cited to establish the publication date of another
     citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or
     other means
"P" document published prior to the international filing date but
     later than the priority date claimed

"T" later document published after the international filing date
     or priority date and not in conflict with the application but
     cited to understand the principle or theory underlying the
     invention
"X" document of particular relevance; the claimed invention
     cannot be considered novel or cannot be considered to
     involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention
     cannot be considered to involve an inventive step when the
     document is combined with one or more other such docu-
     ments, such combination being obvious to a person skilled
     in the art.
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
11 September 2001	25/09/2001

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL – 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Archontopoulos, E

Ir·· ¬national Application No

PCT/US 00/25501

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication,where appropriate, of the relevant passages	Relevant to claim No.
Y	GUMPINGER M.: "GESICHTSMASKE. GUI-BUILDER FACESPAN 3.0 FÜR APPLESCRIPT" CT MAGAZIN FÜR COMPUTER TECHNIK, DE, VERLAG HEINZ HEISE GMBH, HANNOVER, no. 11, 25 May 1998 (1998-05-25), page 66 XP000752489 ISSN: 0724-8679	8-10,15, 16,21, 23,24,29
A	page 66, column 2, line 31 -column 4, line 28 ---	1,14,20, 22,28,30
A	US 5 335 320 A (IWATA MASATAKE ET AL) 2 August 1994 (1994-08-02) abstract column 2, line 30 - line 55 column 5, line 66 -column 12, line 64 -----	1,14,20, 22,28,30

1

INTERNATIONAL SEARCH REPORT

Information on patent family members

Patent document cited in search report	Publication date	Patent family member(s)		Publication date
US 533320 A	02-08-1994	JP	4157528 A	29-05-1992
		JP	7056628 B	14-06-1995

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(54) Title: SYSTEM FOR CREATING DYNAMIC GRAPHICAL USER INTERFACES

(57) Abstract: User interfaces are designed simply, but with a great deal of flexibility by a set of bitmap graphic files. Color codes are predefined for use in a map graphic that specifies the position, size and functionality of control regions based on a correspondence between the color codes and a set of controls that an application program can generate in the user interface. A body graphic defines the overall appearance of the user interface within the size specified by the map graphic. Control graphics define the appearance of the control regions specified by the map graphic based on a predefined order of supplying different states of the control graphics for the control regions. Internal maps within the files containing control graphics may be used to add additional information regarding the size and shape of graphic elements that are displayed in a control graphic, such as the slider on a volume or gain slider control, or the width of characters in a variable pitch font.

AMENDED CLAIMS
[received by the International Bureau on 06 November 2001 (06.11.01);
original claims 1-30 replaced by new claims 1-34 (5 pages)]

1. A method of defining a user interface for a computer program, comprising:

specifying position, size and functionality of at least one control region in a map graphic, by a color of each control region; and

defining appearance of the at least one control region by storing in at least one file a plurality of control graphics, each having a size and shape matching the control region, to be displayed in the at least one control region, the control graphics having a predefined correspondence to different states of user manipulation of an input device.

2. A method as recited in claim 1, wherein said specifying further specifies an extended area of the map graphic for display only in response to a predefined user manipulation of the input device.

3. A method as recited in claim 2, wherein said specifying includes indicating that at least one additional region for at least one of control and information display is located in the extended area.

4. A method as recited in claim 1, wherein the at least one file includes an internal map graphic specifying a size of at least one graphic element also included in the at least one file.

5. A method as recited in claim 1, wherein said specifying includes defining a control region type by whether an internal map graphic for a corresponding control region is stored in the at least one file.

6. A method as recited in claim 5, wherein said defining includes indicating the control region type by the internal map graphic having at least one dimension different than the corresponding control region.

7. A method as recited in claim 6, wherein said defining includes indicating that the plurality of control graphics are stored in the at least one file by the at least one dimension of the internal map graphic being larger than the corresponding control region.

AMENDED SHEET (ARTICLE 19)

8. A method as recited in claim 1, further comprising selecting the map graphic from a plurality of map graphics, each having associated therewith a set of control graphics.

9. A method of defining a user interface for a computer program, comprising:

specifying position, size and functionality of at least one control region in a map graphic, based on a color of each control region; and

dynamically determining a corresponding control graphic for each control region when the corresponding control graphic is displayed.

10. A method as recited in claim 9, wherein the corresponding control graphic is obtained over a network.

11. A method as recited in claim 9, wherein at least one characteristic of the corresponding control graphic is determined by an algorithm.

12. A method as recited in claim 11, wherein the algorithm used to determine the at least one characteristic of the corresponding control graphic includes a randomization function.

13. A computer system to generate a user interface for user interaction with at least one input device and at least one display unit; comprising:

at least one memory unit storing at least one file specifying position, size and functionality of at least one control region in a map graphic, based on a color of each control region, and a plurality of control graphics, each having a size and shape matching the control region and a predefined correspondence to different states of user manipulation of the at least one input device; and

a processor, coupled to the at least one input device, the at least one display unit and said at least one memory unit, to generate the user interface on the at least one display unit based on the map graphic, the plurality of control graphics, and the user manipulation of the at least one input device.

AMENDED SHEET (ARTICLE 19)

14. A computer system as recited in claim 13, wherein said memory unit further stores an extension graphic specifying an extended area of the map graphic for display only in response to a predefined user manipulation of the input device.

15. A computer system as recited in claim 14, wherein said memory unit further stores specification information indicating that at least one additional region for at least one of control and information display is located in the extended area.

16. A computer system as recited in claim 13, wherein said memory unit further stores an internal map graphic specifying a size of at least one graphic element also included in the at least one file.

17. A computer system as recited in claim 13, wherein said memory unit further stores an internal map graphic with at least one dimension different than a corresponding control region to indicate a control region type of the corresponding control region.

18. A computer system as recited in claim 13, wherein said memory unit further stores an internal map graphic with at least one dimension larger than a corresponding control region to indicate that a set of the plurality of control graphics are stored in the at least one file for the corresponding control region.

19. A computer system to generate a user interface for user interaction with at least one input device and at least one display unit, comprising:

at least one memory unit storing at least one file specifying position, size and functionality of at least one control region in a map graphic, based on a color of each control region; and

a processor, coupled to the at least one input device, the at least one display unit and said at least one memory unit, to dynamically determine a corresponding control graphic for each control region when the corresponding control graphic is displayed.

20. A computer system as recited in claim 19, further comprising a network interface to obtain the corresponding control graphic via a network.

21. A computer system as recited in claim 19, wherein said processor is programmed to determine at least one characteristic of the corresponding control graphic using an algorithm.

22. A computer system as recited in claim 21, wherein the algorithm used to determine the at least one characteristic of the corresponding control graphic includes a randomization function.

23. At least one computer readable medium embodying a method of defining a user interface for a computer program, said method comprising:

specifying position, size and functionality of at least one control region in a map graphic, by a color of each control region; and

defining appearance of the at least one control region by storing in at least one file a plurality of control graphics, each having a size and shape matching the control region, to be displayed in the at least one control region, the control graphics having a predefined correspondence to different states of user manipulation of an input device.

24. At least one computer readable medium as recited in claim 23, wherein said specifying further specifies an extended area of the map graphic for display only in response to a predefined user manipulation of the input device.

25. At least one computer readable medium as recited in claim 24, wherein said specifying includes indicating that at least one additional region for at least one of control and information display is located in the extended area.

26. At least one computer readable medium as recited in claim 23, wherein the at least one file includes an internal map graphic specifying a size of at least one graphic element also included in the at least one file.

27. At least one computer readable medium as recited in claim 23, wherein said specifying includes defining a control region type by whether an internal map graphic for a corresponding control region is stored in the at least one file.

AMENDED SHEET (ARTICLE 19)

28. At least one computer readable medium as recited in claim 27, wherein said defining includes indicating the control region type by the internal map graphic having at least one dimension different than the corresponding control region.

29. At least one computer readable medium as recited in claim 28, wherein said defining includes indicating that the plurality of control graphics are stored in the at least one file by the at least one dimension of the internal map graphic being larger than the corresponding control region.

30. At least one computer readable medium as recited in claim 23, further comprising selecting the map graphic from a plurality of map graphics, each having associated therewith a set of control graphics.

31. At least one computer readable medium embodying a method of defining a user interface for a computer program, said method comprising:

specifying position, size and functionality of at least one control region in a map graphic, based on a color of each control region; and

dynamically determining a corresponding control graphic for each control region when the corresponding control graphic is displayed.

32. At least one computer readable medium as recited in claim 31, wherein the corresponding control graphic is obtained over a network.

33. At least one computer readable medium as recited in claim 31, wherein at least one characteristic of the corresponding control graphic is determined by an algorithm.

34. At least one computer readable medium as recited in claim 33, wherein the algorithm used to determine the at least one characteristic of the corresponding control graphic includes a randomization function.

AMENDED SHEET (ARTICLE 19)

This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

❑ **BLACK BORDERS**

❑ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**

❑ **FADED TEXT OR DRAWING**

❑ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**

❑ **SKEWED/SLANTED IMAGES**

☑ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**

❑ **GRAY SCALE DOCUMENTS**

❑ **LINES OR MARKS ON ORIGINAL DOCUMENT**

❑ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**

❑ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.
As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.